

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Christian Zander	Docket No.:	7468 US
Serial No.:	10/692,504	Art Unit:	2179
Filed:	October 24, 2003	Examiner:	Phenuel S. Salomon
For:	Setting up a Procedure of a Communication Taking Place Between Instances Using a Protocol Tester		

MS Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Dear Sir:

This is an appeal from the Examiner's final rejection of the above-identified application as set forth in the Final Office Action dated November 12, 2009 ("Final Action"). A Notice of Appeal was filed in this Application on February 17, 2010.

This brief contains items under the following headings as required by 37 C.F.R. § 41.37(c) and M.P.E.P. § 1205.02:

- | | |
|------|---|
| I. | Real Party In Interest |
| II | Related Appeals and Interferences |
| III. | Status of Claims |
| IV. | Status of Amendments |
| V. | Summary of Claimed Subject Matter |
| VI. | Grounds of Rejection to be Reviewed on Appeal |
| VII. | Argument |
| | Claims Appendix |
| | Evidence Appendix |
| | Related Proceedings Appendix |

I. REAL PARTY IN INTEREST

The real party in interest for this appeal is:

Tektronix, Inc., an Oregon corporation, which is a subsidiary of Danaher Corporation, a Delaware corporation.

II. RELATED APPEALS AND INTERFERENCES

There are no prior or pending appeals, interferences or judicial proceedings known to Appellants, Appellants' legal representative or assignee which may be related to, directly affect or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1-11 are pending in the application and each has been rejected. The rejections of claims 1-11 are being appealed. A Claim Appendix is attached.

The following is the status, as of the date of this submission, of all patented claims:

1. (Pending, twice amended)
2. (Pending, original)
3. (Pending, original)
4. (Pending, original)
5. (Pending, original)
6. (Pending, original)
7. (Pending, original)
8. (Pending, original)
9. (Pending, original)
10. (Pending, original)
11. (Pending, twice amended)

IV. STATUS OF AMENDMENTS

Applicant submitted amendments to claims 1 and 11 on May 28, 2008 and on July 22, 2009. These amendments were entered by the Examiner prior to the Final Action.

V. SUMMARY OF CLAIMED SUBJECT MATTER

A summary of the subject matter defined in each of the independent claims, with reference to page numbers and line numbers, is provided below.

Claim 1 describes a method of setting up a procedure for communication taking place between two instances. There are multiple embodiments. (See, Figure 4 embodiment, at page 5, line 19 to page 6, line 3; Figure 5 embodiment, at page 6, line 4 to page 18; Figure 6 embodiment, at page 6, line 24 - page 7, line 11; and Figure 7 embodiment, at page 7, line 12 to page 9). The steps of the method are executable on a protocol tester. (Page 1, lines 2-4). The method comprises selecting the instances involved in the communication. (Page 4, lines 8-9). A first instance is a protocol tester and a second instance is a device under test. (Page 8, lines 10-13; Figures 4-7, elements TC, IUT). The method further comprises selecting a protocol layer to be emulated on the basis of which the communication between the selected instances is to take place. (Page 4, lines 9-10; page 8, lines 13-14). The method further comprises selecting abstract communication interfaces of the protocol layer which are involved in the communication. (Page 4, lines 10-11, 13-14; page 8, lines 14-16).

The method of Claim 1 further comprises selecting communication data contained in description files to be exchanged at the abstract communication interfaces. (Page 4, lines 14-18; page 8, lines 16, 18-24). The method further comprises setting up a communication procedure executable between the instances through the protocol tester on the basis of the several selecting steps. (Page 4, lines 16-18; page 8, lines 16-18). The selecting steps are performed graphically including a graphic configuration of a communication sequence between the instances involved. (Page 4, lines 16-18; page 8, lines 16-21). The method further comprises defining within the communication data graphically a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable. (Page 3, line 22 - page 4, line 8; page 8, line 21 to page 9, line 2).

The method of 2 depends from claim 1 and further comprises specifying a switch functionality which the other instance executes as a function of the content of the variable. (Page 6, line 23 to page 7, line 11; Figure 6; Annex A1).

The method of 3 depends from claims 1 or 2 and further comprises specifying a loop functionality which the other instance executes as a function of the content of the variable.

(Page 7, lines 12-24; Figure 7; Annex A2).

Claim 11 describes a protocol tester. There are multiple embodiments. (See, Figure 6 embodiment, at page 6, line 24 - page 7, line 11; and Figure 7 embodiment, at page 7, line 12 to page 9). The protocol tester comprises means for selecting instances involved in a communication, with one of the instances being the protocol tester and the other being a device under test. (Page 4, lines 8-9; page 8, lines 10-13; Figures 4-7, elements TC, IUT). The corresponding structure disclosed in the Specification for the “means for selecting instances” is the graphical user interface displays of Figures 6 and 7 and/or a processor running the code recited in Annex A1 or Annex A2, and their equivalents.

The protocol tester further comprises means for selecting a protocol layer to be emulated on the basis of which the communication between the selected instances is to take place. (Page 4, lines 9-10; page 8, lines 13-14). The corresponding structure disclosed in the Specification for the “means for selecting a protocol layer” is the graphical user interface displays of Figures 6 and 7 and/or a processor running the code recited in Annex A1 or Annex A2, and their equivalents.

The protocol tester further comprises means for selecting abstract communication interfaces of the protocol layer which are involved in the communication. (Page 4, lines 10-11, 13-14; page 8, lines 14-16). The corresponding structure disclosed in the Specification for the “means for selecting abstract communication interfaces” is the graphical user interface displays of Figures 6 and 7 and/or a processor running the code recited in Annex A1 or Annex A2, and their equivalents.

The protocol tester further comprises means for selecting communication data contained in description files to be exchanged between the abstract communication interfaces. (Page 4, lines 14-18; page 8, lines 16, 18-24). The corresponding structure disclosed in the Specification for the “means for selecting communication data” is the graphical user interface displays of Figures 6 and 7 and/or a processor running the code recited in Annex A1 or Annex A2, and their equivalents.

The protocol tester further comprises means for automatically setting up a communication procedure that is executable between the instances through the protocol tester on the basis of the selections by the several selecting means. (Page 4, lines 16-18; page 8, lines 16-18). The corresponding structure disclosed in the Specification for the “means for automatically setting up a communication procedure” is the graphical user interface displays

of Figures 6 and 7 and/or a processor running the code recited in Annex A1 or Annex A2, and their equivalents.

The selecting means are graphic selection means and with parameters selectable by the selecting means being used by the setting-up means for setting up the communication procedure that is executable between the instances which makes the setting-up of a communication sequence between the instances involved possible. (Page 4, lines 16-18; page 8, lines 16-21).

The selecting means of the protocol tester further comprises means for defining within the communication data graphically a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable. (Page 3, line 22 - page 4, line 8; page 8, line 21 to page 9, line 2). The corresponding structure disclosed in the Specification for the “means for defining” is the graphical user interface displays of Figures 6 and 7 and/or a processor running the code recited in Annex A1 or Annex A2, and their equivalents.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The Board is requested to consider the following grounds for reversal of the rejection:

(1) Whether the Examiner erred in finding that claims 1-11 are unpatentable over the combination of Gessel, Matsui and Swift.

VII. ARGUMENT

In general, the claimed invention “uses a simplified communication procedure that makes actions of one of two instances of a communication procedure dependent, not on the receipt of events, but on the contents of variables. Rather than branchings being defined over events, a branching is a function of the content of a variable.” (See, Specification page 3, line 22 - page 4, line 2). Switch, loop, jump and/or go-to functionalities may be realized as a function of the content of the variable. (See, page 4, lines 2-8). The advantages of using the variable are obtained by graphically setting up a communication procedure between the protocol tester and device under test. (See, page 4, lines 8-18). Furthermore, a message to be received at the protocol tester from the device under test is graphically defined within the communication data. (See, page 8, line 21-24). The message contains a variable wherein the

protocol tester performs one of several activities as a function of the content of the variable. (See, page 9, lines 1-2).

The applicable legal standards are summarized below:

With respect to alleged obviousness, "an applicant can overcome a [103] rejection by showing insufficient evidence of prima facie obviousness or by rebutting the prima facie case, ..." See, *In re Kahn*, 441 F.3d 977,985-86 (Fed. Cir. 2006). Of course, one cannot show non-obviousness by attacking references individually where the rejection is based on a combination of the references, *In re Keller*, 642 F.2d 413, 416 (CCPA 1981). Rejections based on § 103 must rest on a factual basis with these facts being interpreted without hindsight reconstruction of the invention from the prior art. The Examiner may not "resort to speculation, unfounded assumptions or hindsight reconstruction to supply deficiencies in its factual basis." *In re Warner*, 379 F.2d 10100, 1017 (CCPA 1967), *cert. denied*, 389 U.S. 1057 (1968). An invention "composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art. . . . [I]t can be important to identify a reason that would have prompted a person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does." *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398,418 (2007).

Additionally, "every limitation positively recited in a claim must be given effect in order to determine what subject matter that claim defines." *In re Wilder*, 429 F.2d 447, 450 (CCPA 1970); *See also, In re Wilson*, 424 F. 2d 1382, 1385 (CCPA 1970) ("[a]ll words in a claim must be considered in judging the patentability of that claim against the prior art."). Whether a rejection is correct is determined by first construing the claims and then comparing the properly construed claims to the prior art. *In re Cruciferous Sprout Litigation*, 301 F.3d 1343, 1346 (Fed. Cir. 2002).

A. Comments on the Examiner's remarks in the Advisory Action

In the Final Action, the Examiner cited portions of columns 1 and 2 in Matsui as disclosing a message containing a variable, wherein the protocol tester performs one of several activities as a function of the content of the variable. (Final Action at 4). Columns 1 and 2 in Matsui discuss the system illustrated in Figure 1 of that patent, which is labeled as "Prior Art" to the Matsui disclosure. Appellant objected that the cited portion of Matsui did not disclose or suggest either "variables" or "activities" performed as a function of the

content of variables. (Response Under 37 C.F.R. § 1.116 at 6). In reply, the Examiner failed to explain how columns 1 and 2 taught or suggested the “variable” elements, but instead cited Matsui at columns 5 and 7 and Figure 5 as teaching these elements. (Advisory Action at 2). The cited text at column 5 is related to Matsui’s Figure 2, and the cited text at column 7 is related to Matsui’s Figure 6. The embodiments in Figures 2-8 purport to be embodiments of Matsui’s invention, while the embodiment in Figure 1 is admitted to be the prior art. (Matsui, column 3, line 60 - column 4, line 12). The Matsui embodiments cited in the Advisory Action are not part of the same system as the prior art embodiments cited in the Final Action, and the Advisory Action fails to equate these teachings. Accordingly, the Advisory Action raises new grounds of rejection and effectively admits that the rejections in the Final Action should be withdrawn.

Neither the Matsui invention nor Matsui’s prior art teach or suggest “a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable” as required in independent claims 1 and 11.

B. Grounds 1

1. Claims 1 and 11

Claims 1-11 stand rejected in the Final Action under 35 U.S.C § 103(a) as being anticipated by U.S. Patent No. 5,732,213 to Gessel, et al. (“Gessel”) in view of U.S. Patent No. 6,560,723 to Matsui (“Matsui”) and further in view of WO 98/57268 to Swift, et al. (“Swift”). This rejection is erroneous as the Examiner’s factual findings regarding the teachings of Matsui do not stand up under scrutiny.

Claims 1 and 11 require: defining - or a means for defining - “a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable.” The Final Action admits that the Gessel reference fails to disclose this limitation, but cites the Matsui reference as disclosing these limitations. (Final Action at 3-4, 6). The Swift reference is not cited in connection with these limitations.

In particular, the Final Action cites column 1, lines 58-65 and column 2, lines 10-14 of Matsui in connection with these features. Matsui teaches a test scenario creating unit 150 that describes a test sequence, such as an order of input and output events. (Column 1, lines

58-60). PDU editor 152 specifies a test event to set parameters in an Nth layer PDU, and an ASP editor 153 specifies a test event to set parameters for a lower-level ASP. (Column 1, lines 60-64). The test scenario is used for testing a protocol for a system which implements an Nth layer in an OSI reference model. (Column 2, lines 10-13). Test executing unit 160 tests a system under test on the basis of the created test scenario. (Column 2, lines 13-14).

Matsui discloses creating a test sequence comprising “test events, i.e. inputs and responses.” This test sequence is for use by test executing unit 160. Applicant understands the Final Action to equate the test event “inputs” to the claimed messages that may be received from the device under test. The cited text (column 1, lines 58-65 and column 2, lines 10-14) fails to disclose that the “message to be received . . . from the device under test . . . contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable.” There is simply no discussion of the content or format of Matsui’s “inputs” and no discussion or suggestion regarding variables that might be associated with the Matsui inputs. Instead, Matsui’s test executing unit 160 runs a test sequence, receiving unspecified “inputs” and performing unspecified “responses.” There is no teaching or suggestion in Matsui that a “variable” in the input triggers the protocol tester to “perform one of several activities as a function of the content of the variable” as required in claims 1 and 11.

In the Advisory Action, the Examiner changed the rejection and identified the disclosure at column 5, lines 19-25; item 410 of Figure 5; and column 7, lines 54-58 in Matsui as disclosing the “variable” limitations. No effort was made in the Advisory Action to equate or explain the relationship between this new citation and the Matsui disclosure at columns 1 and 2. Moreover, the new citation also fails to teach or suggest defining - or a means for defining - “a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable.”

The text in column 5 teaches that the Matsui system transmits and receives messages accumulated in buffer 70 in accordance with a sequence description of a test scenario. This text merely discloses exchanging a predetermined series of messages. There is no disclosure of a variable in a message that the protocol tester uses to select one of several activities. The text in column 7 teaches that different types of tests can be created, but also fails to disclose a variable that causes a protocol tester to perform one of several activities.

Finally, element 410 in Figure 5 does not correspond to a message, a variable, or one of several activities performed as a function of the variable. Instead, element 410 is related to a search for test scenarios. Figure 5 illustrates the operation of scenario creating function unit 50/edit searching function 110. (Column 6, lines 35-38). The Advisory Action characterizes element 410 as follows: "Matsui teaches in fig. 5 (item 410) a content of scenario/message with a plurality of scenario and message parameters which are considered as variables" This characterization is clearly erroneous in view of the text of element 410 itself, which reads "What is the content of scenario/message search?" (emphasis added). Element 410 does not disclose content of messages, but instead teaches content of "searches." Without explanation or support, the Examiner equates step 410 to message parameters "which are considered as variables." Column 7, at lines 3-9, makes clear that element 410 is a scenario search step by stating that "At S410, specifically, respective processing is performed such as . . . message parameter search." This disclosure merely reveals that a message may have a "parameter" that be used in a search engine, but fails to teach or suggest that such parameters may be a "variable" wherein "the protocol tester performs one of several activities as a function of the content of the variable."

The Final Action fails to identify: (1) a "variable" taught or suggested in Matsui, and (2) "several activities" performed as a function of the "variable." Thus, claims 1 and 11 are not obvious in view of the proposed combination of Matsui with Gessel and Swift.

2. Claim 2

Claim 2 requires "specifying a switch functionality which the other instance executes as a function of the content of the variable." The Final Action identifies column 7, lines 20-25 of Matsui as teaching this limitation. (Final Action at 4). The cited section of Matsui is copied below.

transmission. Test message input/output function unit **90** is, as an opposite facility to a facility under test, responsible for a function for controlling a test such as start, stop or the like of a simulation test in addition to transmission and reception of a message in accordance with the description of the test scenario. All log information such as scenario execution result is stored in log buffer **100**.

25

The Final Action further characterizes the cited text as follows: “test message function unit that perform transmission and reception based on information content.” (Final Action at 4).

The Matsui disclosure may “perform transmission and reception based on information content” as suggested in the Final Action; however, it does not specify “a switch functionality” that the other instance “executes as a function of the content of the variable” as required in claim 2.

The Matsui disclosure is missing the claimed “variable” as discussed above. Additionally, Matsui does not teach or suggest a specific “switch functionality” that is executed based on “the content of the variable” or any other trigger. The cited text in column 7 merely states that the test message input/output function unit controls the start and stop of a test simulation based on a test scenario description. There is no teaching or suggestion in Matsui that “switch functionality” may be part of a test scenario description.

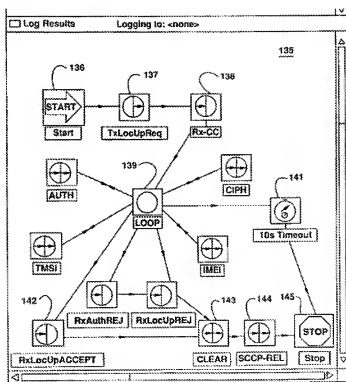
Thus, claim 2 is not obvious in view of the proposed combination of Matsui with Gessel and Swift.

3. Claim 3

Claim 3 requires “specifying a loop functionality which the other instance executes as a function of the content of the variable.” The Final Action identifies column 11, lines 10-15 and element “139” in Figure 13 of Gessel as teaching this limitation. (Final Action at 4-5). The cited sections of Gessel are copied below.

with arrows on both sides of the center vertical line. A
simulated node 139 labeled “Loop” is positioned in the
center of the display and is a holding point where the
simulation waits for another message to be received. A timer
141 monitors the loop for activity, and if there is no activity
for a preset time period (e.g., 10 seconds), the test sequence
is stopped.

Although the cited section of Gessel does, in fact, include the text “loop,” it is not the claimed loop function. Claim 3 requires “loop functionality” that is executed “as a function of the content of the variable.” Gessel discloses a loop function that is part of a simulation and is used by the simulation as a “holding point” at which the simulation “waits” for message. Given this description in Gessel, “LOOP 139” appears to be mislabeled and should be labeled “wait,” “pause,” “hold,” or something similar. The purported loop function in Gessel is not the claimed “loop functionality,” but is simply a step in a test simulation.



LOOP 139 is not executed “as a function of the content of the variable” as required in claim 3. Gessel’s LOOP 139 occurs as part of the test simulation, but is not triggered or executed by a received message. The opposite is actually true - LOOP 139 is waiting for a message. Gessel’s LOOP function occurs before the message and, therefore, before the content of any variable in the message is known.

Thus, claim 3 is not obvious in view of the proposed combination of Gessel with Matsui and Swift.

4. Claims 4-10

Claims 4-10 depend from claim 1 and require additional limitations. Claims 4-10 are rejected under the proposed combination of Gessel, Matsui and Swift. Without conceding that elements of claims 4-10 are present in the Gessel, Matsui or Swift references, Appellant hereby incorporates the above rebuttal arguments by reference. These dependent claims are patentable for at least the same reasons as advanced with respect to claim 1.

C. Conclusion

For all the reasons discussed above, the rejections of claims 1-11 should be reversed.

Respectfully submitted,

May 17, 2010
Date

/Michael J. Fogarty, III/
Michael J. Fogarty, III
Attorney for Appellant
Reg. No. 42,541

FOGARTY, L.L.C.
3010 LBJ Freeway, Suite 1200
Dallas, Texas 75234
214-722-8983
214-272-2778 (fax)

CLAIMS APPENDIX

1. (Previously Presented) A method of setting up a procedure of a communication taking place between two instances, comprising the steps executable on the protocol tester of:

a) selecting the instances involved in the communication, a first instance being a protocol tester and a second instance being a device under test;

b) selecting a protocol layer to be emulated on the basis of which the communication between the selected instances is to take place;

c) selecting abstract communication interfaces of the protocol layer which are involved in the communication;

d) selecting communication data contained in description files to be exchanged at the abstract communication interfaces;

e) setting up a communication procedure executable between the instances through the protocol tester on the basis of the several selecting steps, with the selecting steps being performed graphically including a graphic configuration of a communication sequence between the instances involved; and

f) defining within the communication data graphically a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable.

2. (Original) The method according to claim 1 further comprising the step of specifying a switch functionality which the other instance executes as a function of the content of the variable.

3. (Original) The method according to claims 1 or 2 further comprising the step of specifying a loop functionality which the other instance executes as a function of the content of the variable.
4. (Original) The method according to claim 3 wherein the loop functionality is selected from the group consisting of a for-next, a do-while and a while-do functionality.
5. (Original) The method according to claim 3 further comprising the step of specifying a functionality selected from the group consisting of a jump/go-to functionality and an if-then functionality which the other instance executes as a function of the content of the variable.
6. (Original) The method according to claim 1 wherein the instances involved in the communication are graphically selected, the protocol layer is graphically selected, and the abstract communication interfaces are graphically selected.
7. (Original) The method according to claim 1 wherein the abstract communication interfaces comprise SAPs (Service Access Points).
8. (Original) The method according to claim 1 wherein the communication data comprise data selected from the group consisting of PDUs (Protocol Data Units) and ASPs (Abstract Service Primitives).
9. (Original) The method according to claim 1 wherein the communication data selecting step comprises the steps of: d1) graphically selecting a data format; and d2) graphically setting up the communication sequence between the instances involved.

10. (Original) The method according to claim 9 wherein the communication sequence setting up step comprises the step of entering source code.

11. (Previously Presented) A protocol tester comprising:

a) means for selecting instances involved in a communication, with one of the instances being the protocol tester and the other being a device under test;

b) means for selecting a protocol layer to be emulated on the basis of which the communication between the selected instances is to take place;

c) means for selecting abstract communication interfaces of the protocol layer which are involved in the communication;

d) means for selecting communication data contained in description files to be exchanged between the abstract communication interfaces;

e) means for automatically setting up a communication procedure that is executable between the instances through the protocol tester on the basis of the selections by the several selecting means, which selecting means are graphic selection means and with parameters selectable by the selecting means being used by the setting-up means for setting up the communication procedure that is executable between the instances which makes the setting-up of a communication sequence between the instances involved possible; and means for defining within the communication data graphically a message to be received at the protocol tester from the device under test which contains a variable wherein the protocol tester performs one of several activities as a function of the content of the variable.

EVIDENCE APPENDIX

No evidence was submitted pursuant to §§ 1.130, 1.131, or 1.132 and no other evidence was entered by the Examiner.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings identified in this Brief.